

ou comment installer une fonte casseau (TTF, OT, Type1,...) pour LaTeX en un clic !

par Jacques André

Jacques.AndreNN@orange.fr avec NN=35

Version « de travail » du 8 mars 2010

Résumé

Les casseaux sont des ensembles de caractères souvent disparates (comme les dingbats, les vignettes et lettrines,...) qui se comportent comme des fontes très simples (sans accents, sans ligatures, sans italique ni graisse, etc.). Elles sont souvent en TTF et donc pas directement accessibles en LaTeX.

Cette note présente le programme `ttftotex` qui avec deux paramètres (le nom de la fonte initiale, et un nom de famille « à la Berry ») installe complètement une fonte casseau dans l'arborescence TDS de LaTeX. On décrit aussi divers programmes utilitaires complémentaires.

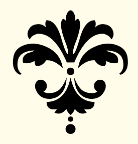
http://jacques-andre.fr/fontex/casseaux.pdf

Sommaire

En général, ce qui est en couleur terre de sienne est « cliquable ».

1 Introduction

S'IL Y A bien quelque chose qui m'agace, c'est qu'avec LaTeX il soit facile de bien positionner des lettrines comme celle de ce début de paragraphe, mais qu'en revanche il soit « compliqué » d'installer la fonte proposant ce bois gravé. De même, si je vois sur le web une fonte offrant des dingbats et ornements, par exemple `DamaskDings1` du site http://www.fontspace.com/nymphont/damask-dings1, il me suffit de la télécharger, de cliquer pour qu'elle entre dans le livre des polices de mon Mac et, en utilisant Word, de sélectionner cette fonte et taper un A pour voir apparaître le cul-de-lampe qui suit, alors qu'avec LaTeX la manip serait bien plus longue.



En revanche, dès qu'on utilise une vraie fonte de texte, par exemple `Utopia`, si ce n'est pas très simple (ni très compliqué) de l'installer pour LaTeX, une fois faite, son utilisation est parfaite : par exemple, on n'a pas à aller chercher les ligatures comme « fi », elles viennent automatiquement.

Mon problème vient bien de là : LaTeX permet de faire des choses très complexes en matière de fontes ; du coup, les auteurs de l'abondante bibliographie sur le sujet (voir page 32) omettent en général de s'étendre sur le cas trivial de ces fontes de dingbats, titrage et autres ornements, que j'appellerai désormais « fontes-casseaux » ; de même les développeurs ne nous ont pas proposé (à ma connaissance) de procédure d'installation simple pour ce cas simple.

Je propose donc ici

- un programme perl qui permet d'installer simplement une fonte-casseau dans l'arborescence TDS de LaTeX,
- un mode d'emploi non moins simple,
- quelques autres procédures connexes (comme par exemple l'affichage de la charte d'une fonte-casseau).

Provisoire... Ce programme (et ceux annexes) et cette documentation sont en version provisoire, pour demander l'avis d'utilisateurs potentiels ou de gourous (voir page 34)...

Limitations actuelles Par ailleurs, pour l'instant, nous supposons
– être sur un Macintosh muni du système OS X, version Léopard (histoire de se distinguer des manuels d'installations souvent implicites pour Windows) ;
– que sont installés `fontforge` et `perl` ;
– nous adresser « à des nuls », dont on demande toutefois qu'ils aient déjà été capables d'installer TEXLive (aujourd'hui 2009), et sachent utiliser *terminal* sur leur Mac. Mais l'adaptation à Windows ou Linux ne devrait poser aucune difficulté !

Enfin, dans ce qui suit, on suppose avoir téléchargé les divers programmes cités depuis `http://jacques-andre.fr/fontex/casseaux.zip` et les avoir copiés dans un répertoire `courant`. De même, la où les fontes à utiliser sont supposées avoir été téléchargées et copiées dans ce même répertoire `courant`.

2 Exemples simples

1. Une fonte TTF – Supposons que nous voulions installer pour LaTeX la fonte `BoltonDropsCaps` qui nous a servi à faire le titre de la couverture de ce document (page 1) et que nous avons téléchargée depuis `http://www.acefreefonts.com/font-10455-Bolton_Drop_Caps.html`.

1. Dans le terminal, depuis le répertoire `courant`, lancer le programme d'installation :
   ```
   perl ttftotex
   ```
2. On vous demande le nom de la fonte :
   ```
   Fonte a installer (forme xxx.ttf ou .pf3, etc.) ?
   ```
   vous répondez alors
   ```
   BoltonDropCaps.ttf
   ```
3. On vous demande un nom LaTeX :
   ```
   Nom LaTeX (forme zxxxx, sans extension) ?
   ```
   En gros (voir détails page 28), c'est un nom de votre invention, de quelques lettres, commençant par « z » et si possible mnémonique. Supposons que vous choisissiez de répondre
   ```
   zbolton
   ```
4. L'installation commence. À un moment, le système va vous demander votre mot de passe pour écrire le fichiers dans l'arborescence TDS. Puis, après quelques lignes de commentaires et un temps parfois très long (!) apparaît le message
   ```
   Fonte zbolton installée - BONS TESTS !
   ```

Lancé depuis n'importe quel répertoire, le programme suivant

```
\documentclass{article}
\usepackage{fourier} % texte courant
\usepackage{zbolton}
\begin{document}
\fontsize{50}{50}\selectfont ABbCdEf4 = \zbolton{ABbCdEf4}
\end{document}
```

doit imprimer :

ABbCdEf4 = ABBCDEF

En effet, l'extension `zbolton.sty` a été créée en même temps que la fonte `zbolton` et placée dans TDS ; elle contient la commande

```
\newcommand{\zbolton}[1]{\fontencoding{U}\fontfamily{zbolton}\selectfont#1}}
```

On note que, pour cette fonte `zbolton`, les majuscules et minuscules ne sont pas différenciées et que le chiffre « 4 » n'a pas de glyphe. On verra (page 21) comment connaître les caractères d'un telle fonte.

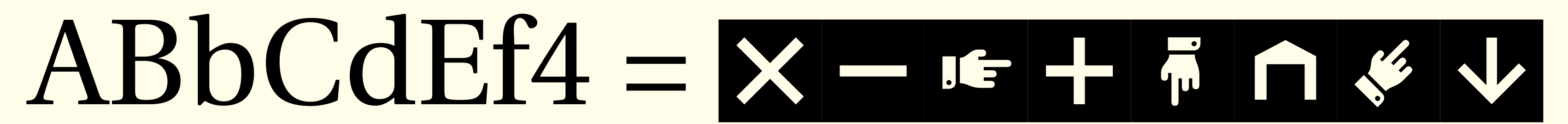
2. Une fonte *Open Type* – Malgré son nom, `ttftotex` permet d'installer tout type de fonte à condition qu'il soit connu de `fontforge` (en fait `ttftotex` signifie « TouTe Fonte to TEX » !) et qu'on ne soit intéressé que par les caractères de numéro < 128_{10}, c'est-à-dire du codage ASCII (voir page 15). À titre d'exemple, on prend une fonte OPEN TYPE de dingbats de *PTF (Porchez Type Foundry)* qui est plus ou moins gratuite : `AllumiPtf-Dingbats.otf`. On a alors le dialogue (dans le terminal) :

```
perl ttftotex
Fonte a installer (forme xxx.ttf ou .pf3, etc.) ? AllumiPtf-Dingbats.otf
Nom LaTeX (forme zxxxx, sans extension) ? zptfding
...
Fonte zptfding installée - BONS TESTS !
```

Cette fois, le programme

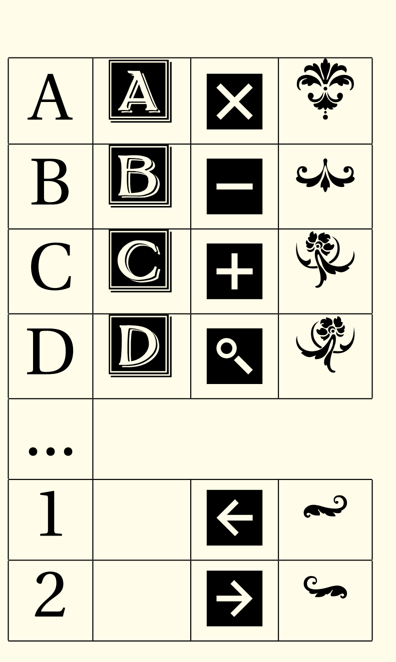
```
\documentclass{article}
\usepackage{fourier} % texte courant
\usepackage{zptfding}
\begin{document}
\fontsize{50}{50}\selectfont ABbCdEf4 = \zptfding{ABbCdEf4}
\end{document}
```

doit imprimer :



3. Plusieurs fontes-casseaux ensemble – Supposons avoir encore installé une troisième fonte, `zdamask`. On peut bien sûr utiliser ces trois fontes ensemble à tout moment dans un programme, comme le suivant, dont le résultat est à sa droite :

```
\documentclass{article}
\usepackage{fourier}\usepackage{zbolton}
\usepackage{zptfding}\usepackage{zdamask}
\begin{document}
\Huge\begin{tabular}{|c|c|c|c|}\hline
A & \zbolton{A} & \zptfding{A} & \zdamask{A}\\\hline
B & \zbolton{B} & \zptfding{B} & \zdamask{B}\\\hline
C & \zbolton{C} & \zptfding{C} & \zdamask{C}\\\hline
D & \zbolton{D} & \zptfding{D} & \zdamask{D}\\\hline
...\\\hline
1 & \zbolton{1} & \zptfding{1} & \zdamask{1}\\\hline
2 & \zbolton{2} & \zptfding{2} & \zdamask{2}\\\hline
\end{tabular}
\end{document}
```



3 Fontes casseaux

Précisons d'abord ce que l'on entend par fonte « casseau ».

Du temps de la typo au plomb, les compositeurs avaient devant eux la casse dont ils se servaient pour le texte courant. Mais ils avaient aussi à portée de main une casse beaucoup plus petite, dite *casseau*, où ils mettaient quelques caractères spéciaux correspondant à un travail souvent spécifique ; ainsi, un éditeur breton y aurait mis un K barré (qui n'existe pas dans Unicode) et un triskell (que l'on ne trouve pas dans les dingbats ordinaires). De même, lorsque la composition nécessitait des lettrines ornées (comme le D de ce début de paragraphe), de vieux symboles comme la feuille aldine ❦, ou des ornements (comme) permettant de faire cadres, frises et autres décorations géométriques, tous ces symboles n'étaient pas dans « la casse » mais dans une autre que par extension on appelle aussi « casseau ».

Ces fontes casseaux avaient quelques propriétés caractéristiques (avec évidemment quelques exceptions) :

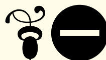
- il n'y avait que peu de signes ;
- ces signes ne correspondaient en général pas à un alphabet ; mais s'il y avait des lettres capitales, on les rangeait dans l'ordre habituel qui est, pour les capitales, l'ordre alphabétique ;
- les concepts de gras, italique, etc. ne s'appliquaient pas à ces signes ;
- le style des signes leur était propre et ne relevaient pas (ou peu) du style des caractères du style courant ;
- etc.

En LaTeX, on peut faire de même : utiliser une « fonte » pour le texte courant et disposer aussi de quelques caractères spéciaux d'une fonte spéciale. Ici, par exemple, j'utilise Fourier-Gutenberg pour le texte courant et par-ci par-là j'utilise quelques caractères pris dans une autre fonte ; par exemple dans le premier paragraphe de cette section, j'ai utilisé la lettrine ornée D et des ornements de Fournier, qui viennent de fontes LaTeX, mais spéciales ; notez que j'ai aussi utilisé la feuille aldine qui est un peu un cas particulier : elle fait partie d'un « casseau » propre à Fourier-GUTenberg [2, 5]. Mais tout utilisateur de LaTeX

peut en utiliser les signes en écrivant par exemple

```
\newcommand{{\futs}[1]{\fontencoding{U}
\fontfamily{futs}\selectfont #1}}
\centerline{\Huge \futs{\char97 A}}
```

pour obtenir les deux caractères



on verra plus bas (voir page 20) qu'à ce casseau est associé un style permettant d'accéder à ces caractères par nom (respectivement `\decotwo` et `\noway`). Nous allons faire ci-après quelques allusions à ce casseau sous son nom LaTeX, `futs`.

Par analogie avec le plomb, on peut dire qu'une fonte casseau LaTeX est caractérisée par quelques propriétés :

- il n'y a que peu de signes ; dans `futs` il y a 32 signes ;
- ces signes ne correspondent en général pas à un codage standardisé (ni Ascii, ni latin-1, ni Unicode,...) ; mais s'il y a de lettres, on les met plutôt selon l'ordre alphabétique ; dans le cas de `futs`, la position du gland (a) et de la feuille de vigne (j) sont complètement arbitraires (et décidées par le dessinateur de la fonte) ;
- les concepts de gras, italique, etc. ne s'appliquent en général pas à ces signes ; une exception pour `futs`, le symbole euro qui peut être €, € ou €, etc.
- on considère ici qu'elles ont comme champ de codage celui de l'Ascii (donc de 0 à 127) ; les lettres accentuées, les ligatures, etc. ne sont pas, à priori, définies ; mais on n'interdit pas LaTeX d'accentuer les « lettres » qui n'en sont pas... par exemple `\futs{ à \'a \c c}` donne ;
- le style des signes leur est propre et ne relève pas (ou peu) du style des caractères du texte courant ; si on conçoit que va mieux avec un caractère du XVIIIe siècle et avec un du XXe siècle, et ne relèvent pas plus du style des caractères d'Utopia (Fourier) que d'Helvetica ou du Times. À plus forte raison pour des dingbats comme ou .
- etc.

Le casseau de Fourier, `futs`, est un peu une exception car on le trouve dans la distribution TeXLive. Mais en général ces fontes casseau n'y sont pas. D'ailleurs certaines ne peuvent pas y être, celles que vous dessinez vous-même (ou composez vous même à partir de glyphes exis-

tants) pour vos propres besoins. Ainsi, ayant eu à utiliser fréquemment les pavages de Truchet (sous forme de tuiles « mi-parties » ou de pavés « coréens » [7]), j'ai été amené à me définir une fonte contenant seulement les huit signes suivants :

Je ne peux pas attendre que TEXLive me fournisse cette fonte et il me faut bien l'installer moi-même !

Quelques fontes casseaux – On en trouve plein sur le Web : il suffit de taper dans Google « *font ornaments initials* » pour voir leur grand nombre, souvent d'ailleurs en *free*. <Liste suivante utile ? ? ?>

Dingbats Par dingbats, on entend toutes ces fontes offrant les dingbats de Zapf, mais aussi bien d'autres, par exemple :
– `zptfding`(bats) que nous citons souvent ici : http://www.typofonderie.com/alphabets/view/AllumiPTF
– ceux de dingbats-depot : http://www.dingbatdepot.com/
– etc.

Initials Lettres ornées, etc. Exemple
– `zbolton` utilisée ici <donner url ?>
– `z1550` utilisée ici

Cadres, bords, coins,. . . Exemple :
– `orna4` utilisée ici

ornements traditionnels Exemple :
– `MonotypeOrnaments 1-6` cf site J.A. ?
– `FellFlower`

ornements de fantaisie Exemple :
– `FairyFantsay` http://www.fontspace.com/windwalker64/wwfairyfantasy

dessins spéciaux Exemple :
– cartes de crédits http://www.fontspace.com/rick-mueller/credit-cards
– panneaux de signalisation http://www.fontspace.com/benn-coifman/roadsign

alphabets spéciaux et bien sûr toute la kyrielles de fontes/polices fantaisies...

4 Mode d'emploi

4.1 Préparatifs

4.1.1 Disponibilité de `fontforge` et de `perl`

Si vous n'en êtes pas sûrs, vérifiez que vous disposez bien de ces deux logiciels, par exemple en tapant dans terminal :

```
which fontforge
which perl
```

si l'un ou l'autre répond « *Command not found* », il vous faut alors le ou les installer (gratuitement). Voir
– pour perl : `http://www.perl.org` ;
– pour fontforge : `http://fontforge.sourceforge.net/`.

4.1.2 Création d'un répertoire de travail `courant`

Nous supposons par la suite travailler dans un répertoire `courant` qui peut s'appeler autrement et être n'importe où, par exemple sur le bureau. Ce répertoire ne sert qu'à en parler ; par ailleurs, on va y créer un certain nombre de fichiers que l'on peut détruire une fois la fonte installée. Il suffit alors de détruire tout le fichier `courant` avec son contenu. <utiliser /TMP/ ?>
Une session commencera donc (dans le terminal) par

```
cd ~/desktop
mkdir courant
cd courant
```

et se terminera par

```
\rm -r -f ~/desktop/courant
```

4.1.3 Téléchargement de la procédure `ttftotex`

Cette procédure et d'autres choses qui vont avec (dont le fichier pdf de ce présent document), sont dans un fichier compressé casseaux.zip qui se trouve à l'url
http://Jacques-Andre.fr/fontex/casseaux.zip
qu'il suffit de télécharger et décompresser (en cliquant par exemple deux fois sur son nom dans le finder). Les divers fichiers apparaissent alors dans le répertoire `casseaux`, à savoir
– `ttftotex` le programme principal,
– `pdftotex` à utiliser lorsque l'on part d'un fichier .pdf (voir page 16),
– `ttftopfb` pour convertir une fonte par exemple ttf en pfb et vérifier sa validité (voir page 16),
– `supcasseau` pour supprimer une fonte casseau de TDS (voir page 16),
– `CasseauCharte.tex` une macro LaTeX, qui imprime la charte d'un casseau (voir page 21).
Il faut copier `ttftotex` dans le répertoire courant : `cp .../Casseau/ttftotex ~/desktop/courant`

4.1.4 Téléchargement de la fonte, et éventuel changement de nom

On suppose que le fonte choisie a été téléchargée et décompressée. Elle se présente alors en général sous forme d'un fichier dont le suffixe est .ttf, .pfb, .ot, etc. Reprenons le premier exemple (page 5) : depuis http://www.acefreefonts.com/font-10455-Bolton_Drop_Caps.html, on télécharge un dossier `Bolton_Drop_Caps.zip`, on clique dessus deux fois et on obtient le fichier `BoltonDropCaps.ttf` (en général il y a plusieurs fichiers regroupés dans un répertoire). Il faut le copier dans le répertoire courant :

```
cp .../BoltonDropCaps.ttf ~/desktop/courant
```

Compatibilité des noms de fontes avec Unix, Linux, LaTeX, etc. Certaines fontes ont parfois des noms pouvant poser problème ici par la suite à cause de la présence de certains caractères comme « _ » (par exemple `uesbi___.pfb`), voire des espaces (comme `Orna 4.ttf`). Pour éviter tout problème, je recommande de les renommer (après éventuelles duplication) directement dans le finder. Ce renommage n'a aucune implication sur leur contenu ! Pour les deux exemples précédents, on préférera `uesbi.pfb` et `Orna4.ttf`.
Absence de suffixe Il arrive fréquemment que les fontes n'ont pas de suffixe. Par exemple la fonte ayant servi à faire le cadre de la couverture de ce document, a été téléchargée depuis `http://www.urbanfonts.com/dingbats/borders-dingbats.htm` et a donné un répertoire `ornam4` qui contient une fonte dont le nom est « `Orna 4` », sans suffixe. Or, on en a besoin d'un par la suite, il faut donc en ajouter : il suffit de cliquer 1 fois sur ce nom dans le finder pour voir qu'il s'agit d'une *true type font* dont le suffixe normal est `.ttf`. On l'ajoute donc (par le finder) au nom de la fonte qui devient `Orna 4.ttf`. On reconnaît le nom avec blanc que vous venons de traiter. On fait donc les deux opérations d'un coup et renommons cette fonte directement `Orna4.ttf`

Dans tous les cas, on doit avoir dans le répertoire `courant` un nom de fonte correct (sans espace, _, ...) avec son suffixe. On continue ici nos exemples avec `Orna4.ttf` qu'on a donc copié dans le répertoire `courant` par `cp .../ornam4/Orna4.ttf ~/desktop/courant`

4.2 Appel de `ttftotex`

En fait, le mode d'emploi de notre procédure a été donné dans les exemples de la section 2. Voici toutefois quelques précisions sur des cas particuliers.
Nous supposons à nouveau que notre procédure `ttftotex` a été mise dans le répertoire `courant` ainsi que la fonte à installer (nous prenons ici comme exemple `Orna4.ttf`).
Le déroulement de l'installation de la fonte dans l'arborescence TDS se déroule comme suit :

1. Dans le terminal, depuis le répertoire `courant`, lancer le programme d'installation :
   ```
   perl ttftotex
   ```
2. On vous demande le nom de la fonte :
   ```
   Fonte a installer (forme xxx.ttf ou .pf3, etc.) ?
   ```

vous répondez alors `Orna4.ttf`
C'est `fontforge` qui vérifiera, un peu plus tard, qu'il s'agit bien d'une fonte correcte. En revanche, on vérifie que la fonte est dans le répertoire courant, sinon on abandonne.

3. On vous demande un nom LaTeX :

   ```
   Nom LaTeX (forme zxxxx, sans extension) ?
   ```

   En gros (voir détails page 28), c'est un nom de votre invention, de quelques lettres, commençant par « z » et si possible mnémonique. Dans notre cas, on tape par exemple `zorn4`
   Le programme vérifie alors qu'il n'y a pas déjà une fonte de ce nom installée parmi les fontes-casseaux. Si c'est le cas, on propose soit de SUPPRIMER l'ancienne version (c'est-à-dire de faire `updmap-sys --disable zorn4.map`, ce qui prend un certain temps) et de poursuivre pour installer la nouvelle fonte du même nom (zorn4), soit de taper un nouveau nom (et alors on revérifie que ce n'est pas encore le nom d'une fonte déjà installée), soit de taper STOP pour arrêter le programme.

4. `fontforge` est appelé pour convertir la fonte en .pfb. Si vous aviez déjà un fichier .pfb, vous auriez pu partir d'ici. Voir le programme `pfbtotex` (page 16).
   Si la fonte ne convient pas, `fontforge` dit pourquoi et on arrête le programme.
   Si il y a des problèmes mineurs, `fontforge` émet des notes dans le *log* et l'installation continue, à partir des deux fichiers .pfb et .afm créés par `fontforge`.

   Lors de la conversion de la fonte en .pfb, on réduit le codage à celui de Latin-1. En effet, même si on trouve de plus en plus de fontes codées théoriquement avec Unicode, en général seul... Ascii est utilisé.

5. L'installation pour TeX à proprement parler commence alors. À un moment, le système va vous demander votre mot de passe pour écrire le fichiers dans l'arborescence TDS. Puis, après de nombreuses lignes de commentaires et « un certain temps » (il faudrait supprimer l'option *quiet* pour montrer que le programme n'est pas planté comme on pourrait le croire vu la lenteur de `updmap`), doit apparaître le message

```
Fonte zbolton installée - BONS TESTS !
```

Mais on peut aussi avoir, et c'est le cas pour `zorn4`, un message comme

```
ATTENTION : il y a des chiffres dans le nom zorn4 : on garde le nom zorn4.sty
mais on y a change les  commandes \newcommand{\zorn4...} en \newcommand{\zornd...}
```

En effet, si pour TEX `zorn4.sty` est un nom correct malgré la présence d'un chiffre, il n'en est pas de même pour `\zorn4`. On change donc d'autorité les noms en y remplaçant les chiffres 12...9 respectivement par ab...i et 0 par o.
N.B. La commande `\zorn4` serait l'équivalente normale de `\zbolton` (voir page 6), mais pour utiliser le caractère de même code que A de la fonte `zorn4` il faut écrire `\zornd{A}` comme annoncé par le message ci-dessus, et non `\zorn4{A}`. L'utilisateur peut toujours modifier ce nom dans le style correspondant, qui est à l'adresse :
`/usr/local/texlive/texmf-local/tex/latex/casseaux/zorn4.sty`

4.3 Autres procédures

pfbtotex Fonctionne comme `ttftotex`, mais on part d'un fichier `xxx.pfb` *et* d'un fichier `xxx.afm`. `fontforge` n'est pas appelé.

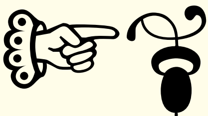
ttftopfb Commence comme `ttftotex` : lit la fonte, la transforme en `.pfb` (et `.afm`) si elle est valide, vérifie que le nom proposé pour le casseau `zzzz` ne correspond pas déjà à une fonte casseau existante dans TDS. Mais la fonte n'est pas installée. On peut le faire ensuite avec `pfbtotex`.

supcasseau Supprime de l'arborescence TDS une fonte casseau (suppression des fichiers .pfb, .fd, .sty, etc. et recalcul de la map).

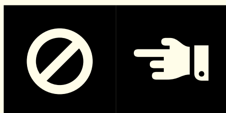
5 Utilisation d'une fonte casseau

5.1 Utilisation « normale »

Pour utiliser des caractères de la fonte `futs (Fourier-Ornaments)`, on peut écrire par exemple `{\fontencoding{U}\fontfamily{futs}\fontsize{50}{50}\selectfont ta}` pour obtenir



de même, la fonte `AllumiPtf-Dingbats.otf` que nous avons installée dans TDS sous le nom de `zptfding` (voir page 7) peut être utilisée de la même façon :
`{\fontencoding{U}\fontfamily{zptfding}\fontsize{50}{50}\selectfont ta}` donne



La grande difficulté avec ces fontes est de savoir à quel glyphe de la fonte correspond un code (par exemple `\char97`) ou un caractère (par exemple `a`). On verra (page 21) la façon de construire une charte, ou table d'équivalence de ces fontes.

Attention : le nom du codage, donné par `\fontencoding{U}` (U voulant dire *Unknown*, inconnu, signifie en fait « ni T1, ni OT1, etc. »), est important car sinon LaTeX va chercher une fonte du même type que celle englobante que notre installation ne prévoit justement pas. On aurait ici (où on a du Fourier en T1), sans cette indication de codage U, un message du type

```
Font shape ‘T1/zptfding/m/n’ undefined
using ‘T1/cmr/m/n’ instead on input line nnn.
```

et la sortie montrerait les deux lettres ta en cmr !
En revanche, on définit les fontes casseaux sans extension de forme ou graisse. Ce qui veut dire que

la graisse (`\fontseries`) est toujours moyenne (m) et que la forme (`\fontshape`) est romain normal (n). Les trois expressions suivantes :

```
{\fontencoding{U}\fontfamily{zptfding}\fontsize{50}{50}\selectfont ta}
{\fontencoding{U}\fontfamily{zptfding}\fontshape{it}\fontsize{50}{50}\selectfont ta}
{\fontencoding{U}\fontfamily{zptfding}\fontsize{50}{50}\selectfont \textbf{ta}}
```

donnent donc la même chose (toutefois les deux dernières produisent un message dans le log). Enfin, la mention `\fontsize{50}{50}` peut être supprimée : on prend alors la force de corps courant.

5.2 Fichier `.sty`

L'écriture de choses comme `{\fontencoding{U}\fontfamily{zptfding}\fontsize{12}{12}\selectfont X}` (même si on fait l'impasse de `\fontsize`) est plutôt fastidieuse et on a donc envie de faire une macro

```
\newcommand{\zptfding}[1]{{\fontencoding{U}\fontfamily{zptfding}\selectfont #1}}
```

(justement sans `fontsize`). L'endroit idéal pour la mettre est dans un style associé à la fonte. C'est pourquoi en installant une fonte casseau ZZZZ on installe aussi un fichier ZZZZ.sty qui est le suivant :

```
%Filename: ZZZZ.sty
\NeedsTeXFormat{LaTeX2e}%}
\ProvidesPackage{ZZZZ}%
%
\newcommand*{\ZZZZ}[1]{{%
\fontencoding{U}\fontfamily{ZZZZ}\selectfont  #1}}
%
\endinput
```

où ZZZZ est remplacé par le nom de la fonte-casseau installée (par exemple `zptfding`). Tous ces fichiers .sty sont placés dans l'arborescence TDS, plus précisément dans

```
/usr/local/texlive/texmf-local/tex/latex/casseaux/ZZZZ.sty
```

Comme on l'a déjà vu (page 16), on n'accepte que des noms légaux de commande, ce qui oblige à changer les chiffres en lettres (l'utilisateur peut donc toujours corriger le fichier .sty comme il le veut).

Les instructions suivantes

```
\usepackage{zptfding}
...
\begin{center}\Huge

a) {\fontencoding{U}\fontfamily{zptfding}\selectfont u b w}\\
b) \zptfding{u b w}

\end{center}
```

donnent donc

a)

b)

5.3 Appel des caractères par nom

Lorsqu'il y a peu de caractères dans un casseau et lorsque ceux-ci sont caractérisables, il peut être agréable de leur donner un nom. C'est par exemple ce que fait la fonte `futs (Fourier-Ornaments)` qui a des déclarations du type

```
\newcommand*{\aldine}{{\fontencoding{U}\fontfamily{futs}\selectfont \char111}}
\newcommand*{\lefthand}{{\fontencoding{U}\fontfamily{futs}\selectfont \char116})
```

Si on déclaré l'extension `fourier` ou même seulement `fourier-orns`, il suffit alors d'écrire `\aldine\lefthand` pour avoir ❦☞. Voir la liste complète dans [2].

Notre programme `ttftotex` n'a aucune possibilité de décider de ces noms (même en regardant l'afm, car souvent les casseaux gardent les noms « A, B, ... »). On recommande simplement à chaque utilisateur qui aimerait disposer de tels noms de les mettre dans son fichier .sty ! <en dire plus ?>

À noter que cette méthode ne s'applique pas à tous les casseaux, par exemple pas à ceux formés d'ornements difficiles à dénommer, comme (il s'agit d'un extrait des fontes Monotypes Ornaments – voir `http://jacques-andre.fr/fontex/ornements-charte.pdf`) ; des notations comme `\decoone, \decotwo, \decothree...` devraient aller jusqu'à `\decotwohundredandfifty` au moins, alors autant se rappeler que le caractère `\char97` correspond à ✽. . .D'où l'utilité des chartes de la section suivante !

5.4 Charte d'un casseau

Il est important de connaître l'équivalence entre une touche (disons un caractère ASCII) et le glyphe lui correspondant dans une fonte-casseau.
On peut bien sûr se contenter de la procédure standard TeX `nfssfont.tex` décrite dans le LaTeX *Companion* [15, page 377] et dont le source est à l'url `/usr/local/texlive/2007/texmf-dist/tex/latex/base/nfssfont.tex`.
Lancé dans le terminal (par `latex nfssfont.tex`) on a par exemple le dialogue suivant :

```
**********************************************
* NFSS font test program version <v2.2b>
*
* Follow the instructions
**********************************************

Input external font name, e.g., cmr10
(or <enter> for NFSS classification of font):

\currfontname=z1550
Now type a test command (\help for help):)
*\table\bye
[1]
Output written on nfssfont.dvi (1 page, 3808 bytes).
```

Il suffit de faire `dvips` puis `ps2pdf` pour obtenir `nfssfont.pdf` (si on procédait depuis `TeXshop`, le résultat y serait direct). Le résultat est en figure 1.

	´0	´1	´2	´3	´4	´5	´6	´7	
´04x									"2x
´05x									
´06x									"3x
´07x									
´10x		A	B	C	D	E	F	G	"4x
´11x	H	I	J	K	L	M	N	O	
´12x	P	Q	R	S	T	V	V	W	"5x
´13x	X	Y	Z						
´24x									"Ax
´25x		©							
	"8	"9	"A	"B	"C	"D	"E	"F	

FIGURE 1 – Sortie de `nfssfont.tex` pour `z1550`

Ne trouvant pas ce tableau très lisible, on donne les outils pour un construire un plus complet, avec par ligne les codes décimaux et hexa, le caractère « clavier » correspondant au caractère « casseau », la chasse et la hauteur totale de chaque caractère du casseau ; chaque ligne n'est imprimée que pour les codes existants ; par ailleurs pour des raisons de lisibilité, le caractère casseau est dans un corps choisi par l'utilisateur (les chasse et hauteur sont relatives à ce corps).

Il suffit alors de copier `CharteCasseau.tex` (depuis notre paquet `casseaux.zip`) dans un répertoire où on écrit le programme suivant

```
\documentclass{article}
\usepackage{z1550} % z1550 ou toute autre fonte-casseau
\input{CharteCasseau}
\begin{document}
\CharteCasseau{z1550}{24} % idem ; 24 = corps des caractères du casseau
\end{document}
```

pour obtenir la liste

Déc.	Hexa.	Car.	z1550	chasse	hauteur
32	20	␣		4.79999pt	0.0pt
35	23	#		22.32pt	23.016pt
45	2D	-		24.0pt	12.14398pt
48	30	0		23.16pt	22.89601pt
49	31	1		22.79999pt	23.208pt
50	32	2		24.0pt	21.432pt
51	33	3		24.0pt	21.432pt
52	34	4		24.0pt	14.424pt
53	35	5		24.0pt	7.22401pt
54	36	6		24.0pt	14.496pt
55	37	7		24.0pt	7.22401pt
56	38	8		24.0pt	14.13599pt
57	39	9		24.0pt	14.328pt
65	41	A		24.0pt	23.42401pt

Déc.	Hexa.	Car.	z1550	chasse	hauteur
66	42	B	B	24.0pt	23.496pt
67	43	C	C	24.0pt	22.72798pt
68	44	D	D	24.0pt	23.37599pt
69	45	E	E	24.0pt	23.25598pt
70	46	F	F	24.0pt	22.79999pt
71	47	G	G	24.0pt	23.136pt
72	48	H	H	24.0pt	23.42401pt
73	49	I	I	24.0pt	23.28pt
74	4A	J	I	24.0pt	23.28pt
75	4B	K	K	24.0pt	23.23201pt
76	4C	L	L	24.0pt	23.23201pt
77	4D	M	M	24.0pt	23.64001pt
78	4E	N	N	24.0pt	23.59201pt
79	4F	O	O	24.0pt	23.688pt
80	50	P	P	24.0pt	24.072pt
81	51	Q	Q	24.0pt	23.59201pt
82	52	R	R	24.0pt	23.952pt

Déc.	Hexa.	Car.	z1550	chasse	hauteur
83	53	S		24.0pt	23.28pt
84	54	T		24.0pt	23.088pt
85	55	U		24.0pt	23.4pt
86	56	V		24.0pt	23.4pt
87	57	W		24.0pt	23.42401pt
88	58	X		24.0pt	23.352pt
89	59	Y		24.0pt	23.04pt
90	5A	Z		24.0pt	23.64001pt
95	5F	_		24.0pt	11.37599pt

Enfin je propose aussi de mettre dans ce paquet les routines pour présenter les fontes casseaux selon le clavier lorsque ce dernier est utilisé pour placer côte à côte des caractères liés. Par exemple la fonte `RococoOrnamentsThree` de Monotype met à la place de 7 8 9 les caractères ╔ ═ ╗ utiles pour faire des hautes de cadres, et de même en 7 u j ceux pour le côté gauche et en 9 o k ceux pour le droit, etc. Tout ceci apparaît bien dans la figure 2 où on voit aussi le matériel pour des cadres du type ⌐||||||||||¬ et ╔══╗

Avis ?

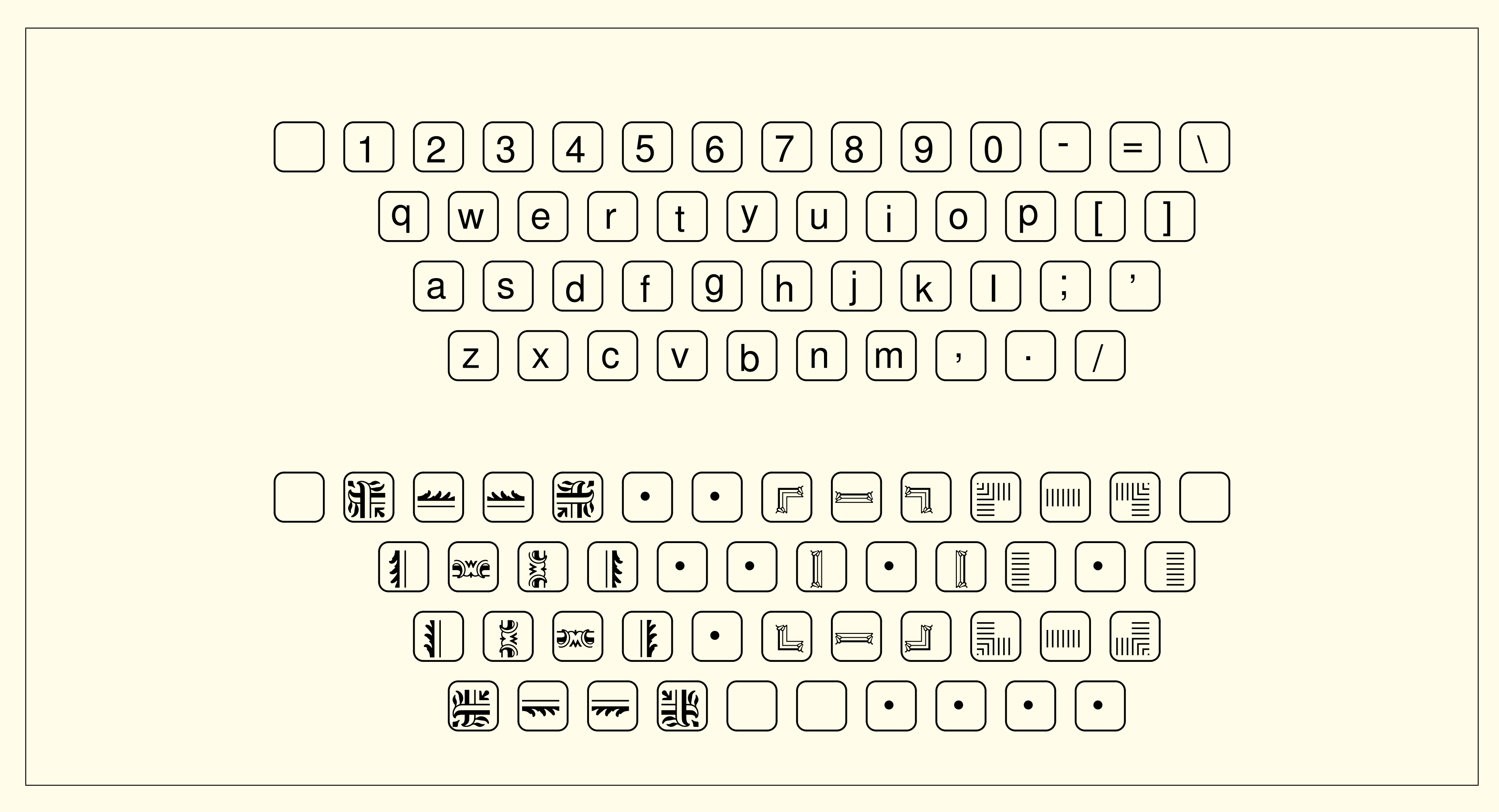


FIGURE 2 – Ornements de la fonte-casseau RococoOrnamentsThreeMT (`mpi006`).

6 Principes adoptés

La simplicité des procédures proposées ici vient en grande partie de l'adoption de principes… simplificateurs (mais non réducteurs) :

6.1 Arborescence

1. On respecte l'utilisation de la branche `texmf-local` pour les données propres à chaque installation locale. C'est dans cette dernière branche, non modifiée par les futures versions de TEXlive, que l'on place normalement les fontes « personnelles ».
2. Selon [13], on classe normalement les fontes comme suit : les divers fichiers (`.tfm, .afm, pfb, etc.`sont dans des répertoire spécialisés (`tfm, afm, type1, etc.`) mais ensuite chacun ventilé selon sa fonderie d'origine : les afm des fontes d'Adobe sont dans le répertoire `fonts/afm/adobe`, celles de Monotype dans `fonts/afm/monotype` etc. (voir figure 3).

   On n'a pas adopté ce principe, ou plutôt on a considéré que toutes ces fontes viennent d'une même et unique fonderie, appellée `casseaux`. En effet, ces fontes casseaux sont souvent créées par des individus isolés et on on a presque autant de fonderies différentes que de fontes ; par ailleurs, de nombreuses fontes-casseaux sont vendues par plusieurs fonderies. Enfin, chaque fonte casseau n'a qu'un fichier de chaque type (un seul fichier .pfb par exemple) alors qu'une fonte de texte peut en avoir des dizaines ! Enfin, ça nous évite de demander à l'utilisateur un nom de fonderie, ce qui simplifie la construction automatique de cette arborescence.
3. Selon Karl Berry [3, 9], les noms « internes » des fontes sont liées au nom de la fonderie. Par exemple le Times et l'Helvetica d'Adobe sont notés respectivement `ptm` et `phv`, `p` étant le « code » d'Adobe. De même Fourier-GUTenberg, basé sur Utopia d'Adobe, est appelé `put`.

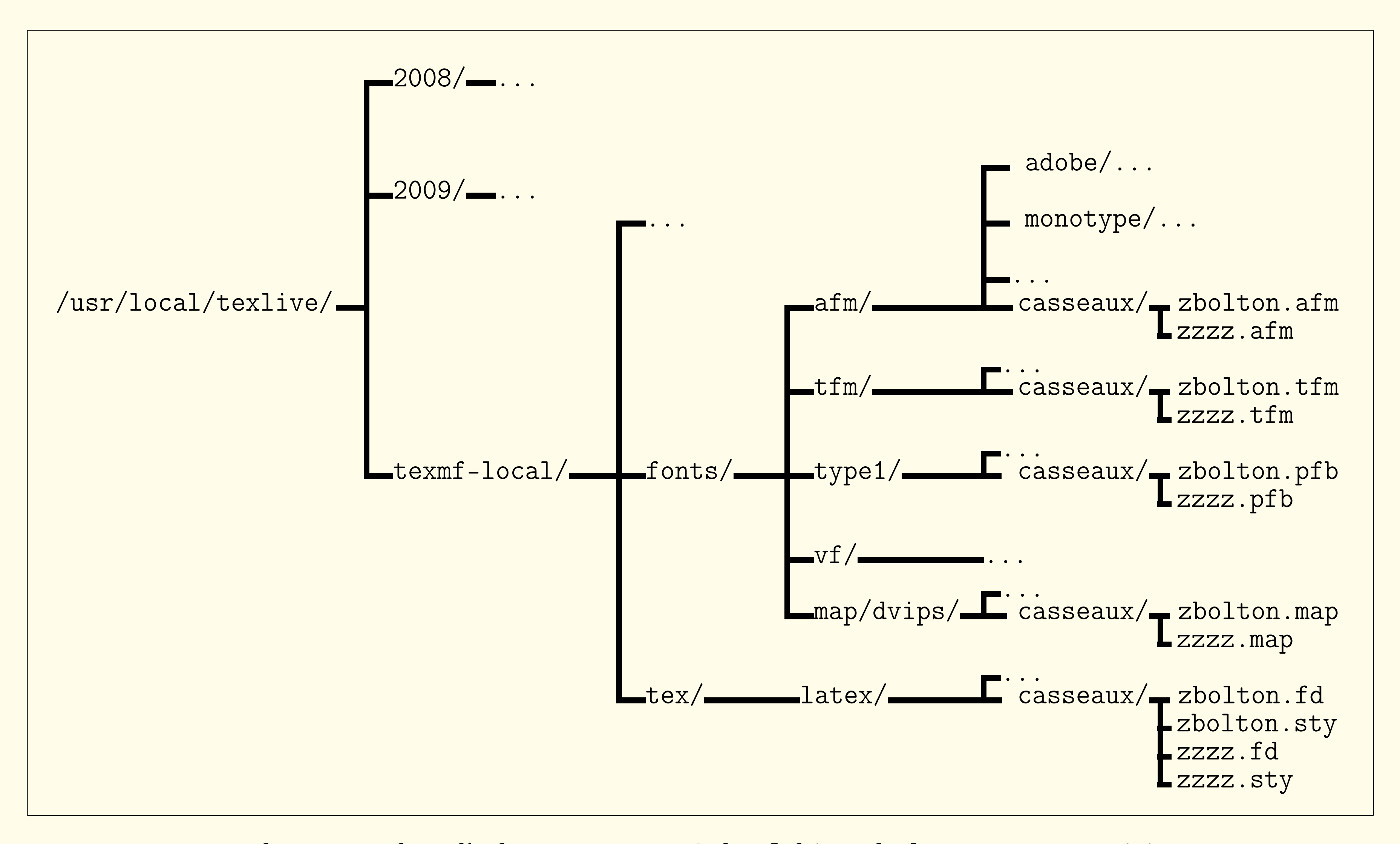


FIGURE 3 – Emplacement dans l'arborescence TDS des fichiers de fontes casseaux (ici on suppose en avoir installé deux : `zbolton` et `zzzz`).

En fait Berry a prévu une catégorie spéciale pour les autres fonderies, dont le nom commence alors par « z » (sans doute pour les « zautres fonderies » ;-). Sans obliger l'utilisateur de notre procédure d'installation d'adopter ce nomage des fontes par z, on le recommande donc, mais on laisse le reste des autres lettres au choix de l'utilisateur de façon que ce soit un minimum mnémonique ou proche de la fonte d'origine. !

6.2 Installation de la fonte

Si on installe une fonte-casseau avec la machinerie habituelle (notamment `fontinst`), on peut faire des constatations intéressantes :

1. On voit que les fichiers construits sont très peu nombreux et très courts : un seul fichier `.tfm`, un seul fichier `.pfb`, etc. et certains fichiers sont complètement inutiles (comme ceux pour les fontes virtuelles `vf`).
2. ces fichiers sont en général très courts, ; par exemple les fichiers de type .fd n'ont que quelques lignes seulement

   ```
   \DeclareFontFamily{U}{zxxxx}{}
   \DeclareFontShape{U}{zxxxxx}{m}{n}{ <-> zxxxxx}{}
   ```

   tandis que les fichiers `map` se réduisent à

   ```
   zxxxx FontName <zxxxx.pfb
   ```

3. etc.

Or, on se rend vite compte qu'on a toutes les informations en main pour créer soi-même ces fichiers, sans passer par `fontinst`. Seul `updmap` est utilisé !
Perl (comme d'autres langages de ce type il est vrai, mais c'est celui que j'utilisais naguère) permet de faire facilement ces diverses manipulations de noms de fichiers, de substitutions, tout en accédant au système, etc.

7 Problèmes rémanents

Comme je l'ai dit, ce programme et ses confrères sont encore en cours de validation. Toutefois certains problèmes sont encore à résoudre (dans un ordre quelconque) :

1. Cet ensemble n'a été testé que sur MacOSX, avec pdftex.
2. Il faudrait l'adapter à Windows ?
3. Mon programme perl n'est sans doute pas écrit selon un méthode très canonique (je date de de l'époque d'Algol 60 ;-)
4. Comment faire pour en faire un package LaTeX ?
5. Je ne comprend pas pourquoi `updmap` est aussi lent, même en version *quiet*.

Pour toute suggestion, etc., voir page 34.

Références

[1] William ADAMS, « Font Installation : Agfa Eaglefeather to Linotype Zapfino », *TUGboat*, Volume 22, Proceedings of the 2001 Annual Meeting, nº 3, p. 247-250.

[2] Jacques ANDRÉ, Les ornements de la fonte Fourier-GUTenberg, http://jacques-andre.fr/fontex/Fourier-orn.pdf, 2009.

[3] Karl BERRY,*Fontname, Filenames for TEX fonts*, september 2005, http://tug.org/fontname/fontname.pdf

[4] Thierry BOUCHE, « Tout savoir sur les fontes avec LATEX », *Tutoriel des Journées GUT'99*, 18 mai 1999, ancienne url = http://www.gutenberg.eu.org/pub/GUTenberg/manifestations/gut99/fontes/tutoriel.pdf

[5] Michel BOVANI, *Fourier-GUTenberg*, mars 2004, fourier-doc-fr.pdf. Se trouve sur les DVD TEXLive >2007. Accessible aussi par ftp : télécharger depuis ftp://ftp.gutenberg.eu.org/pub/gut/distribs/fourier/fourier-v1.3 fourier-GUT.tgz, le « dézipper » et prendre fourier-GUT/doc/latex/fourier/fourier-doc-fr.pdf.

[6] Bernard DESGRAUPES, *LATEX, apprentissage, guide et référence*, 2e édition, Vuibert, Paris, 2000.

[7] Philippe ESPERET et Denis GIROU, « Coloriage du pavage de Truchet », *Cahier GUTenberg*, nº 31, décembre 1998, p. 5-18 ; http://cahiers.gutenberg.eu.org/cg-bin/article/CG_1998___31_5_0.pdf

[8] Daniel FLIPO, *Administration d'une distribution TEX*, http://daniel.flipo.free.fr/doc/tex-admin/TeX-admin.pdf.

[9] *Filenames for TEX fonts* http://www.tug.org/fontname/html/

[10] Maarten GELDERMAN, « How to install a Type1 Font using fontinst », *MAPS*, 22, Vorjaar 1999, p. 37-42. http://www.ntg.nl/maps/22/11.pdf

[11] Ferdy HANSSEN, « Installing fonts in LaTeX : a user's experience », *MAPS*, 29, Najaar 2003, p. 61-64. http://www.ntg.nl/maps/29/13.pdf.

[12] Yannis HARALAMBOUS, *Fontes et codages*, O'Reilly, Paris, 2004.

[13] Alan JEFFREY et coll., *fontinst : Font installation software for TeX*, http://www.tug.org/applications/fontinst/

[14] Philipp LEHMANN, *The font installation guide - Using PostScript fonts to their full potential*, http://www.ctan.org/tex-archive/info/Type1fonts/fontinstallationguide/.

[15] Frank MITTELBACH, Michel GOOSSENS et autres, *LaTeX Companion, 2e édition*, Pearson Education, Paris, 2005.

[16] « TDS : une structure de répertoires pour les fichiers TeX », Traduction de Jean-Côme Charpentier et introduction de Fabrice Popineau, *Cahiers GUTenberg*, nº 44-45, novembre 2004, p. 113-114. http://cahiers.gutenberg.eu.org/cg-bin/article/CG_2004___44-45_83_0.pdf

[17] Bruno VOISIN (with help from Bob Kerstetter), *Installing Fonts for LaTeX on OS X*, http://www.tug.org/mactex/fonts/fonttutorial-current.html, janvier 2004.

Pour toute critique, réclamation, suggestion, demande de renseignement…

Jacques.AndreNN@orange.fr avec NN=35



http://Jacques-Andre.fr/fontex/casseaux.pdf